Name:

ID:

CSCI 3104, Algorithms
Exam 2 – S18

Profs. Chen & Grochow
Spring 2020, CU-Boulder

Instructions: This quiz is open book and open note. You **may** post clarification questions to Piazza, with the understanding that you may not receive an answer in time and posting does count towards your time limit (30 min for 1x, 37.5 min for 1.5x, 45 min for 2x). Questions posted to Piazza **must be posted as PRIVATE QUESTIONS.** Other use of the internet, including searching for answers or posting to sites like Chegg, is strictly prohibited. Violations of these are grounds to receive a 0 on this quiz. Proofs should be written in **complete sentences. Show and justify all work to receive full credit.**

YOU MUST SIGN THE HONOR PLEDGE. Your quiz will otherwise not be graded. Honor Pledge: On my honor, I have not used any outside resources (other than my notes and book), nor have I given any help to anyone completing this assignment.

Your Name: ____________________

Standard 18. Suppose we have n stairs to climb. You may choose to jump up either $1, 2$, or 3 stairs. Your goal is to count the number of ways to climb the stairs. Note that your starting position is on the ground floor and not on the first stair.

Is there a clear recursive structure in the problem that would be useful in designing an effective dynamic programming algorithm? That is, is dynamic programming a useful algorithmic technique for this problem? Clearly justify your answer.

Solution: Suppose that we jump i steps forwards, where $i \in \{1, 2, 3\}$. Then we have a new instance of the stair climbing problem, with $n - i$ stairs as the initial size of our new sub-problem. In particular, we observe that the sub-problems overlap, which we can leverage to store the results for a linear number of subproblems in our lookup table.

We note that the number of ways to climb the stairs may be enumerated by the following recurrence:

$$f(n) = \begin{cases} n & : n = 1, 2, \\ 4 & : n = 3, \\ f(n-1) + f(n-2) + f(n-3) & : n > 3. \end{cases}$$